What is claimed is:

1.   A method comprising:

5   receiving a user selection of a first instruction from a

list of instructions that executed during a processor

simulation; and

        tracing an operand in the first instruction directly to

a use of the operand in a second instruction in the list of

10   instructions by following operand dependencies between such

first and second instructions.

2.   The method of claim 1 wherein tracing determines that the

second instruction set the value of the operand as used in the

15   first instruction as a source operand.

3.   The method of claim 1 wherein tracing determines that a

next use of the operand, after that of the first instruction

as a destination operand, occurs in the second instruction.

20

4.   The method of claim 1 wherein tracing comprises:

        determining attributes of the first instruction; and

        using the attributes of the first instruction to find the

second instruction.

5.   The method of claim 4 wherein receiving comprises:

receiving a selected cycle corresponding to the first instruction.

5

6.   The method of claim 5 further comprising:

determining a program counter value associated with the selected cycle.

10 7.   The method of claim 6 wherein determining attributes comprises:

using the program counter value to look up the attributes in an instruction operand map that provides attributes of each instruction, including instruction type and type of registers used by such instruction type for operands.

8.   The method of claim 7 wherein the instructions are instructions of a microcode and the instruction operand map is generated at microcode build time.

20

9.   The method of claim 7 wherein using the attributes comprises:

determining for each register type a physical address.

10. The method of claim 9 wherein determining the physical address comprises determining whether each register type is a non-I/O register or an I/O register.

5

11. The method of claim 10 wherein determining the physical address comprises determining whether each non-I/O register is accessed using an index register.

10 12. The method of claim 11 wherein the instruction operand map is used to provide the physical address for each non-I/O register that is not accessed using an index register.

13. The method of claim 11 wherein the physical address for

15 each non-I/O register that is determined to be accessed using an index register is determined by obtaining a historical value of the index register at the selected cycle from a register history that records historical values of registers for each register type as such values change during

20 simulation.

14. The method of claim 12 wherein the physical register for any register determined to be an I/O register is obtained for

51

the selected cycle from a memory reference history that

records physical addresses and reference counts for each of

the I/O registers that is used in a memory reference during

simulation.

5

15. The method of claim 9 wherein determining the program

counter value comprises looking up the program counter value

in a program counter history that records state change events,

which are detected during simulation, with associated program

10 counter values for each cycle in which such state change

events occurred.

16. The method of claim 15 wherein tracing further comprises:

using the physical address for each register used in the

15 first instruction to traverse the program counter history,

instruction by instruction, to find a matching physical

address in the second instruction.

17. The method of claim 16 wherein the microcode is intended

20 for execution on one or more microengines in a processor

simulated by the processor simulation and wherein the program

counter history of more than one of the microengines is

traversed.

18. The method of claim 1 wherein the instructions are intended for execution on at least one microengine of the processor simulated by the processor simulation.

5

19. The method of claim 18 wherein the microengine is configured to support multiple threads of execution and the microcode is intended for execution by at least one of the multiple execution threads.

10

20. An article comprising:

a storage medium having stored thereon instructions that when executed by a machine result in the following:

receiving a user selection of a first instruction from a

15   list of instructions that executed during a processor simulation; and

tracing an operand used in the first instruction directly to a use of the operand in a second instruction in the list of instructions by following operand dependencies between such

20   first and second instructions.

21. The article of claim 20 wherein tracing determines that the second instruction set the value of the operand as used in the first instruction.

5   22. The article of claim 20 wherein tracing determines that a next use of the operand after that of the first instruction occurs in the second instruction.

23. A graphical user interface comprising:

10   a window showing a view of microcode instructions that executed on a processor simulator during a simulation and for which a simulation history has been collected by the processor simulator;

        the view being usable to provide a tracing option in a

15   menu presented to a user for one of the instructions as an instruction of interest; and

        the tracing option being usable to trace any variable used by the instruction of interest in the simulation history directly to a second instruction in which a most recent change

20   to or next use of such variable occurred.

24. The graphical user interface of claim 23 wherein selection of the tracing option by the user causes a submenu

of options available for the instruction of interest to be

provided to the user, each of the options of the submenu

corresponding to one of the variables used by the instruction

of interest.

5

25. The graphical user interface of claim 22 further

comprising:

a second window in which a cycle of interest

corresponding to the instruction of interest is indicated;

10   wherein the indication of the cycle of interest is

modified to indicate a new cycle of interest corresponding to

the second instruction; and

wherein the first window is modified to reflect the new

cycle of interest.

15

26. A device comprising:

at least one line card for forwarding networking data to

ports of a switching fabric;

the at least one line card comprising a network processor

20 comprising multi-threaded microengines each configured for

execution with a microcode; and

wherein the microcode comprises a microcode developed

using a debugger tool that allowed tracing of operands in code

lines of the microcode once executed by a simulator simulating operation of the network processor.

27.   The system of claim 26 wherein the operands are associated with registers in the microengines, and the registers include general purpose registers and I/O transfer registers.